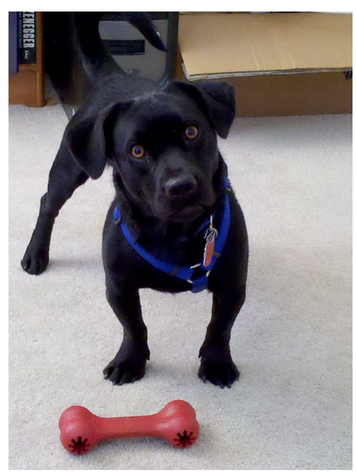
## CSCI211: Intro Objectives

- Introduction to Algorithms, Analysis
- Course summary
- Reviewing proof techniques



## My Bio

- From Dallastown, PA
- B.S., Gettysburg College
- M.S., Duke University
- Ph.D., University of Delaware
- For fun: pop culture, gardening, volunteer at Rockbridge Animal Alliance



# What This Course Is About


From
*30 Rock*



Now, everything comes down to expert knowledge of **algorithms** and **data structures**.
If you don't speak fluent **O-notation**, you may have trouble getting your next job at the technology companies in the forefront.

-- Larry Freeman

For me, great algorithms are the poetry of computation. Just like verse, they can be terse, allusive, dense, and even mysterious. But once unlocked, they cast a **brilliant new light** on some aspect of computing.

-- *Francis Sullivan*

## Motivation Google

- From a Google interview preparation email

Get your algorithms straight (they may comprise up to a **third** of your interview).
Visit: http://en.wikipedia.org/wiki/List_of_algorithm_general_topics
and examine this list of algorithms:
http://en.wikipedia.org/wiki/List_of_algorithms
and data structures: http://en.wikipedia.org/wiki/List_of_data_structures
Write out all the algorithms yourself from start to finish and make sure they're working.



## What is an Algorithm?

- Precise procedure to solve a problem
- Completes in a finite number of steps

## Questions to Consider

- What are our goals when designing algorithms?
- How do we know when we've met our goals?

> - Goals: Correctness, Efficiency
> - Use analysis to show/prove



## Course Goals

- Learn how to formulate precise problem descriptions
- Learn specific algorithm design techniques and how to apply them
- Learn how to analyze algorithms for efficiency and for correctness
- Learn when no exact, efficient solution is possible

## Course Content

- Algorithm analysis
  - Formal – proofs; Asymptotic bounds
- Advanced data structures
  - e.g., heaps, graphs
- Greedy Algorithms
- Divide and Conquer
- Dynamic Programming
- Network Flow
- Computational Intractability



## Course Notes

- Textbook: *Algorithm Design*
- Participation is encouraged
  - Individual, group, class
- Assignments:
  - Reading text, writing brief summaries
    - Readings through Friday due following Monday
  - Solutions to problems
  - Analysis of solutions
  - Programming (little)

Given on Friday, due next Friday

## Course Grading

- 38% Individual written and programming homework assignments
- 30% Two midterm exams
- 20% Final
- 7% Text book reading summaries, weekly
  - In a journal on wiki
- 5% Participation and attendance



## Journal Content

- Brief summary of chapter/section
  - ~1 paragraph of about 5-10 sentences/section; feel free to write more if that will help you
- Include motivations for the given problem, as appropriate
- For algorithms, brief sketch of algorithm, intuition, and implementation
  - Include runtime
- Questions you have about motivation/solution/proofs/analysis
- Discuss anything that makes more sense after reading it again, after it was presented in class (or vice versa)
- Anything that you want to remember, anything that will help you
- Say something about how readable/interesting the section was on scale of 1 to 10

# Journal Grading

| Grade | Meaning |
|---|---|
| ✔+ | Especially well-done, insightful questions |
| ✔ | Typical grade |
| ✔- | Unsatisfactory write up; will have specific feedback |
| 0 | No submission |



# ALGORITHMS

## Computational Problem Solving 101

- Computational Problem
  - A problem that can be solved by logic
- To solve the problem:
  1. Create a *model* of the problem
  2. Design an *algorithm* for solving the problem using the model
  3. Write a *program* that implements the algorithm



## Computational Problem Solving 101

- Algorithm: a well-defined recipe for solving a problem
  - Has a finite number of steps
  - Completes in a finite amount of time
- Program
  - An algorithm written in a programming language
  - Important to consider implementation’s effect on runtime

# PROOFS



## Why Proofs?

- What are insufficient alternatives?
- How can we prove something isn't true?

# Why Proofs?

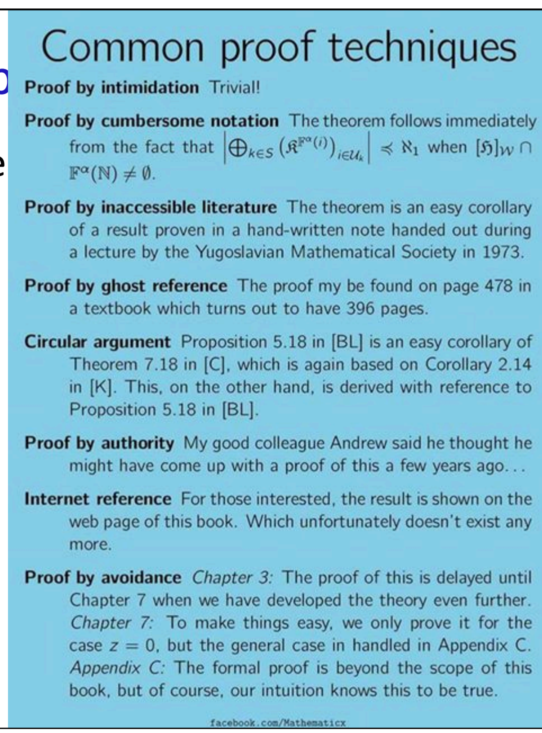
- What are insufficient alternatives?
  - ➢ Examples
    - Considered all possible?
  - ➢ Empirical/statistical evidence
    - Ex: "Lying" with statistics
- How can we prove something isn't true?
  - ➢ One counterexample

Need irrefutable proof that something is true—for ***all*** possibilities



# Soap Op

- "It's the



## Common proof techniques

**Proof by intimidation** Trivial!

**Proof by cumbersome notation** The theorem follows immediately from the fact that $\left|\bigoplus_{k\in S}\left(\mathfrak{K}^{\mathbb{F}^{\alpha}(i)}\right)_{i\in\mathcal{U}_k}\right| \preccurlyeq \aleph_1$ when $[\mathfrak{H}]_{\mathcal{W}} \cap \mathbb{F}^{\alpha}(\mathbb{N}) \neq \emptyset$.

**Proof by inaccessible literature** The theorem is an easy corollary of a result proven in a hand-written note handed out during a lecture by the Yugoslavian Mathematical Society in 1973.

**Proof by ghost reference** The proof my be found on page 478 in a textbook which turns out to have 396 pages.

**Circular argument** Proposition 5.18 in [BL] is an easy corollary of Theorem 7.18 in [C], which is again based on Corollary 2.14 in [K]. This, on the other hand, is derived with reference to Proposition 5.18 in [BL].

**Proof by authority** My good colleague Andrew said he thought he might have come up with a proof of this a few years ago. . .

**Internet reference** For those interested, the result is shown on the web page of this book. Which unfortunately doesn't exist any more.

**Proof by avoidance** *Chapter 3:* The proof of this is delayed until Chapter 7 when we have developed the theory even further. *Chapter 7:* To make things easy, we only prove it for the case $z = 0$, but the general case in handled in Appendix C. *Appendix C:* The formal proof is beyond the scope of this book, but of course, our intuition knows this to be true.

facebook.com/Mathematicx

Not discussed in class

From Joel Feinstein
University of Nottingham
"Why do we do proofs"

## Analyzing Statistics

Two hospitals (A and B) each claim to be better at treating a certain disease than the other.

**Hospital A**

- cured a greater % of its *male patients* last year than Hospital B
- cured a greater % of its *female patients* last year than Hospital B

**Hospital B**

- cured a greater % of its *patients* last year than Hospital A

Given that none of the #s involved are zero, is it possible that both hospitals have their calculations correct?
If so, which hospital would you rather be treated by?



Not discussed in class

From Joel Feinstein
University of Nottingham
"Why do we do proofs"

## Example

| Hospital | Male Patients | % | Female Patients | % | Total Patients | % |
|---|---|---|---|---|---|---|
| A | 50/100 | 50% | 1/1 | 100% | 51/101 | 50.5% |
| B | 24/50 | 48% | 49/50 | 98% | 73/100 | 73% |

Well-known phenomenon: Simpson's Paradox

# Common Types of Proofs?



# Common Types of Proofs

- Direct proofs
  - Series of true statements, each implies the next
- Proof by contradiction
- Proof by induction

## Proof By Contradiction

What are the steps to a proof by contradiction?



## Proof By Contradiction

1. Assume the proposition (P) we want to prove is false
2. Reason to a contradiction
3. Conclude that P must therefore be true

## Prove: There are Infinitely Many Primes



## Prove: There are Infinitely Many Primes

- What is a prime number?
- What is not-a-prime number?

- What is our first step (proof by contradiction)?
- What do we want to show?

## Prove: There are Infinitely Many Primes

- Assume there are a finite number of prime numbers
  - List them: $p_1$, $p_2$ ..., $p_n$
- Consider the number $q = p_1p_2... p_n + 1$

What are the possibilities for q?

q is either composite or prime



## Prove: There are Infinitely Many Primes

- Assume there are a finite number of prime numbers
  - List them: $p_1$, $p_2$ ..., $p_n$
- Consider the number $q = p_1p_2... p_n + 1$
- Case: q is composite
  - If we divide q by any of the primes, we get a remainder of 1 → q is not composite

## Prove: There are Infinitely Many Primes

- Assume there are a finite number of prime numbers
  - List them: $p_1$, $p_2$ ..., $p_n$
- Consider the number $q = p_1p_2\ldots\, p_n + 1$
- Case: q is composite
  - If we divide q by any of the primes, we get a remainder of 1 → q is not composite
- Therefore, q is prime, but q is larger than any of the finitely enumerated prime numbers listed → *Contradiction*

Proof thanks to Euclid



## Proof By Induction

What are the steps to a proof by induction?

# Proof By Induction

1. What you want to prove
2. Base case
   - Typical: Show statement holds for $n = 0$ or $n = 1$
3. ***Induction hypothesis***
4. Induction step: show that adding one to n also holds true
   - Relies on earlier assumptions

When/why is induction useful?

Show true for all (infinite) possibilities
Show works for "one more"



# Proof By Induction

1. State your P(n).
   - P(n) is a property as a function of *n*
     - State for which *n* you will prove your P(n) to be true
2. State your base case.
   - State for which n your base case is true, and prove it
     - Use the smallest *n* for which your statement is true
3. State your induction hypothesis
   - Without an induction hypothesis, the proof falls apart.
   - Usually it is just restating your P(n), with no restriction on *n* (an arbitrary n)
4. Inductive Step.
   - Consider P(n + 1).
     - Try to prove a larger case of the problem than you assumed in your induction hypothesis.
   - Keep in mind: What are you trying to prove?
   - Use your induction hypothesis, and clearly state where it is used. If you haven't used your induction hypothesis, then you are not doing a proof by induction.
5. Conclusion.
   - Optionally, restate the problem.

## Example of Induction Proof

**Prove**:
2+4+6+8+... + 2n = n*(n+1)



## Example of Induction Proof

**Prove**:
2+4+6+8+... + 2n = n*(n+1)

For what values of *n* do we want to prove this is true?

A: where *n* is a natural number

## Example of Induction Proof

**Prove**: 2+4+6+8+… + 2n = n*(n+1)

(where *n* is a natural number)

- **Base case**: *n* = 1 →
  - 2*1 = 1*(1+1) ✔



## Example of Induction Proof

**Prove**: 2+4+6+8+… + 2n = n*(n+1)

(where *n* is a natural number)

- Base case: *n* = 1 →
  - 2*1 = 1*(1+1) ✔
- **Induction Hypothesis:**
  - Assume statement is true for some arbitrary k > 1

# Example of Induction Proof

**Prove:** 2+4+6+8+… + 2n = n*(n+1)

(where *n* is a natural number)

- Base case: *n* = 1 →
  - 2*1 = 1*(1+1) ✔
- Induction Hypothesis:
  - Assume statement is true for some arbitrary k > 1
- Prove holds for *k*+1



# Example of Induction Proof

**Prove:** 2+4+6+8+… + 2n = n*(n+1)

(where *n* is a natural number)

- Base case: *n* = 1 →
  - 2*1 = 1*(1+1) ✔
- Induction Hypothesis:
  - Assume statement is true for some arbitrary k > 1
- Prove holds for *k*+1, i.e., show that 2+4+6+8+… + 2k + 2(k+1) = (k+1)*((k+1)+1)

**Prove:** 2+4+6+8+… + 2n = n*(n+1)

- Base case: *n* = 1 → 2*1 = 1*(1+1) ✔
- Assume statement is true for arbitrary *n*=*k*>1
- Prove true for *k*+1, i.e., show that 2+4+6+8+… + 2k + 2(k+1) = (k+1)*((k+1)+1)
  - 2+4+6+8+… + 2k + 2(k+1)

  = k*(k+1) + 2(k+1)

  = $k^2$ + k + 2k + 1

  = $k^2$ + 3k + 1

  = (k+1)*(k+2)

  = (k+1)*((k+1)+1) ✔

Approach shown: transform LHS to RHS

I want to see these steps in your proofs!



**Prove:** 2+4+6+8+… + 2n = n*(n+1)

- Base case: *n* = 1 → 2*1 = 1*(1+1) ✔
- Assume statement is true for arbitrary *n*=*k*>1
- Prove true for *k*+1, i.e., show that 2+4+6+8+… + 2k + 2(k+1) = (k+1)*((k+1)+1)
  - 2+4+6+8+… + 2k + 2(k+1)

  = k*(k+1) + 2(k+1)

  = (k+1)*(k+2), factor out the (k+1)

  = (k+1)*((k+1)+1) ✔

Alternative solution

## Proof Summary

- Need to ***prove*** conjectures
- Common types of proofs
  - Direct proofs
  - Contradiction
  - Induction
- Common error: not checking/proving assumptions
  - "Jumps" in logic



## Proof: All Horses Are The Same Color

- **Base case**: If there is only *one* horse, there is only one color.
- **Induction step**: Assume as induction hypothesis that within any set of $n$ horses, there is only one color.
  - Look at any set of $n + 1$ horses
  - Label the horses: 1, 2, 3, ..., $n$, $n + 1$
  - Consider the sets $\{1, 2, 3, ..., n\}$ and $\{2, 3, 4, ..., n + 1\}$
  - Each is a set of only $n$ horses, therefore within each there is only one color
  - Since the two sets overlap, there must be only one color among all $n + 1$ horses

Where is the error in the proof?

## Error in Proof

- **Base case**: If there is only *one* horse, there is only one color.
- **Induction step**: Assume as induction hypothesis that within any set of *n* horses, there is only one color.
  - Look at any set of *n* + 1 horses
  - Number them: 1, 2, 3, ..., *n*, *n* + 1
  - Consider the sets {1, 2, 3, ..., *n*} and {2, 3, 4, ..., *n* + 1}
  - Each is a set of only *n* horses, therefore within each there is only one color
  - *Since the two sets overlap*, there must be only one color among all *n* + 1 horses

Does not hold true when n+1=2

**Lesson**: check assumptions within proof



## Looking Ahead

- Check out course wiki page
  - Test username/password
  - Decide which style of journal you want: wiki or blog
- Read first two pages of book's preface
  - Summarize on Wiki by next Tuesday @ midnight